

What is claimed is:

1. A system comprising:
a central processor;
a non-volatile memory coupled with the central processor and storing platform firmware; and
a machine-readable medium coupled with the central processor, the machine-readable medium to be used in initializing an operating environment for the system upon power up, the machine-readable medium comprising a first set of instructions forming at least a portion of the operating environment, and a second set of instructions defining one or more firmware extensions to enable the system to access the first set of instructions.
2. The system of claim 1, wherein the machine-readable medium comprises a hard disk platter.
3. The system of claim 2, wherein the one or more firmware extensions comprise a file system driver to support a file system format not supported by the platform firmware.
4. The system of claim 1, wherein the non-volatile memory comprises random access non-volatile memory.
5. The system of claim 1, wherein the central

processor comprises a central processing unit housed in a single chip.

6. The system of claim 5, further comprising:

a volatile memory; and

a motherboard coupling the volatile memory, the non-volatile memory and the machine-readable medium with the central processing unit.

7. A machine-readable medium comprising:

a first set of instructions defining operations for enabling a machine to access a second set of instructions comprising at least a portion of an operating system stored on the machine-readable medium in a format that is unreadable by the machine before the machine loads the first set of instructions; and

the second set of instructions.

8. The machine-readable medium of claim 7, wherein the first set of instructions comprise one or more extensions to platform firmware capability.

9. The machine-readable medium of claim 8, wherein the portion of an operating system comprises an operating system loader.

10. The machine-readable medium of claim 9, wherein the one or more extensions to platform firmware capability comprise a file system driver to support a file system format used to store at least a portion of the second set of instructions.

11. The machine-readable medium of claim 9, wherein the one or more extensions to platform firmware capability comprise glyphs that represent a language.

12. The machine-readable medium of claim 9, wherein the one or more extensions to platform firmware capability comprise a Unicode collation module.

13. A machine-implemented method for extending platform firmware capabilities, the method comprising:

loading one or more firmware extensions from a boot media;

booting the system; and

loading and running an operating system loader from the boot media using the one or more loaded firmware extensions.

14. The machine-implemented method of claim 13, wherein loading one or more firmware extensions from a boot

media during a system boot comprises using a block input/output protocol to abstract a mass storage device containing the boot media.

15. The machine-implemented method of claim 14, wherein the one or more firmware extensions comprise a file system driver to support a file system format used to store data on the boot media.

16. The machine-implemented method of claim 15, wherein the one or more firmware extensions further comprise glyphs that represent a language.

17. A data processing system comprising:
means for processing instructions and data;
non-volatile memory means for storing platform firmware; and
mass storage means providing means for extending platform firmware capabilities during system booting before an operating system loader is loaded and run.

18. The system of claim 17, wherein the mass storage means comprises an optical disk.

19. The system of claim 18, wherein the means for

extending platform firmware capabilities comprise a file system driver to support a file system format not supported by the platform firmware.

20. The system of claim 19, wherein the non-volatile memory means comprises random access non-volatile memory.